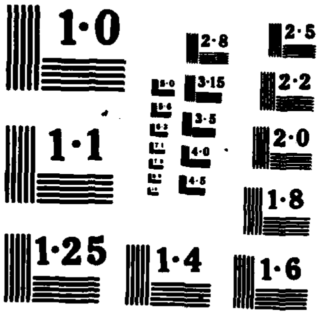
AD A157 451 INTEROPERABILITY TEST MESSAGE DATA BASE(U) ARMY ELECTRONIC PROVING GROUND FORT HUACHUCA AZ R G JACQUES SEP 84 1/1

UNCLASSIFIED F/G 9/2 NL

END
DATE
FILMED
10 85


1·0
2·8
2·5
3·15
2·2
3·5
1·1
4·0
2·0
4·5
1·8
1·25
1·4
1·6

2

UNCLASSIFIED

DDC AD NUMBER ________

FUNDING PROJECT NO. IT665702D625

TECOM PROJECT (TRMS) NO. 7-CO-RD4-EPO-003

TEST ACTIVITY REPORT NO. ________

TEST SPONSOR: US ARMY TEST AND EVALUATION COMMAND

METHODOLOGY INVESTIGATION

FINAL REPORT

INTEROPERABILITY TEST MESSAGE DATA BASE

by

Richard G. Jacques

September 1984

US ARMY ELECTRONIC PROVING GROUND
FORT HUACHUCA, ARIZONA

UNCLASSIFIED

DISPOSITION INSTRUCTIONS

Destroy this report in accordance with appropriate regulations when no longer needed. Do not return it to the originator.

DISCLAIMER

Information and data contained in this document are based on input available at the time of preparation. Because the results may be subject to change, this document should not be construed to represent the official position of the US Army Materiel Command unless so stated.

The use of trade names in this report does not constitute an official indorsement or approval of the use of such commercial hardware or software. This report may not be cited for purposes of advertisement.

DEPARTMENT OF THE ARMY
HEADQUARTERS, U.S. ARMY TEST AND EVALUATION COMMAND
ABERDEEN PROVING GROUND, MARYLAND 21005 - 5055

REPLY TO
ATTENTION OF

AMSTE-AD-M

SUBJECT: Methodology Investigation Final Report on Interoperability Test Message Data Base, TRMS No. 7-CO-RD4-EPO-003

Commander
U.S. Army Electronic Proving Ground
ATTN: STEEP-MT-DA
Fort Huachuca, AZ 85613-7110

1. Subject report is approved.

2. Test for the Best.

FOR THE COMMANDER:

GROVER H. SHELTON
C, Meth Imprv Div
Analysis Directorate

A-1

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
TRMS No. 7-CO-RD4-EPO-003

2. GOVT ACCESSION NO.
A154461

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
METHODOLOGY INVESTIGATION FINAL REPORT INTEROPERABILITY TEST MESSAGE DATA BASE

5. TYPE OF REPORT & PERIOD COVERED
Methodology Investigation Final Report

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Richard G. Jacques

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
US Army Electronic Proving Ground
Fort Huachuca, AZ 85613

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
1T665702D625

11. CONTROLLING OFFICE NAME AND ADDRESS
US Army Test and Evaluation Command
Attn: DRSTE-AD-M
Aberdeen Proving Ground

12. REPORT DATE
September 1984

13. NUMBER OF PAGES
62

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Data Base Management System
Message Format Library
Test Item Stimulator
Test Scenario

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The procedures and mechanism for defining a message format library have been developed and validated for the Test Item Stimulator (TIS). The use of a commercially available relational DBMS was found to be a viable alternative for developing a data base of test messages. The data base schema, high-level menu-driver, and procedures developed were validated using the character-oriented Maneuver Control System (MCS) as a test case. → see p3

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

PAGE

SECTION 1. SUMMARY

Paragraph Number

SECTION 2. DETAILS OF INVESTIGATION

SECTION 3. APPENDICES

LIST OF FIGURES

LIST OF TABLES

FOREWORD

Ultrasystems Technology, Incorporated, Sierra Vista, Arizona
assisted in the preparation of this document under
Contract Number DAEA18-83-C-0003.

1.0 SUMMARY

1.1 Background

a. The Interoperability Test Message Data Base (ITMDB) investigation was initiated to provide a method for developing, using, and maintaining an automated data base of digital stimulus messages and valid response messages between each system under test (SUT) and all other systems with which it must interoperate for both technical and tactical scenarios.

b. The Department of Defense (DOD) has developed, and is continuing to develop, a number of automated Command, Control, Communication, and Intelligence (C^3I) systems. Although each of these automated systems has a different function and a different set of requirements, the automated systems all use digital message exchange to communicate. Testing to verify that these automated systems meet their operational requirements is largely an exercise of each system's software implementation as measured by the output message stream.

c. In the past, the verification and validation of software has been accomplished by a highly individualized type of testing usually done by the software developers. Individual modules of software are tested, then systems are tested independently. However, no comprehensive test methodology has been available to verify the interoperability of the systems as a whole. This has resulted in unreliable products when integrated. Because the U.S. Army of the 1990's will depend upon the systems developed now, an orderly, rigorous testing methodology has been developed to augment the software testing procedure.

d. The Interim Test Item Stimulator (ITIS) was a test driver which was developed by the U.S. Army Electronic Proving Ground (USAEPG) for developmental testing (DT) of the Maneuver Control System (MCS). The ITIS proved useful for single-system testing with prescripted message streams. Interoperability testing, by definition, requires the ability to simultaneously test multiple systems and allow for the real-time generation and insertion of messages into test message streams. Thus, the ITIS has evolved into the Test Item Stimulator (TIS) to meet these additional test requirements. Test conduct using the TIS is separated functionally into three phases: pre-test scenario preparation, real-time item stimulation, and post-test data reduction and analysis.

(1) The first phase is the pre-test generation of test cases that will sufficiently test the system. To adequately stimulate these systems for test, each specific message type needs to be generated, inserted into a data base, and composed into technical and tactical scenarios. For performance and stress testing, the scenarios may contain thousands of messages. The capability of not only generating correct messages, but also the capability to control and inject known errors is required (e.g., sync loss, parity, timeouts).

(2) The second phase is the real-time test. This is the conduct of the test as determined by the test director in the pre-test phase. The TIS stimulates the SUT by transmission of prescripted messages. The results of the real-time test are recorded for use in the third phase, the post-test analysis.

(3) An in-depth study of the test results is conducted during the post-test analysis phase. This is the phase wherein the SUT performance is measured against the requirements. The results of this analysis will become the substance of a test report for those tests dealing with software functionality on a system level.

1.2 Objective

The Interoperability Test Message Data Base investigation was initiated to provide a method for developing, using, and maintaining an automated data base of digital stimulus messages and valid response messages between each system and all other systems with which it must interoperate for both technical and tactical scenarios.

1.3 Summary of Procedures

The objective of this methodology investigation was accomplished through the following steps.

a. The technical requirements and the set of operations which must be performed in order to create and maintain a data base of message formats and scenario data were determined. The various types of data required were identified, as well as the functional requirements.

b. Based on the requirements analysis, a tradeoff study was done to determine the suitability of various commercial data base management system (DBMS) products.

c. A data base schema and a high-level menu-driven interface to the data base functions were developed in support of the TIS.

d. Procedures were outlined for defining a set of message formats and the interfaces used to edit these messages.

e. A subset of the MCS messages was used to verify the procedures outlined for defining character-oriented message formats.

1.4 Summary of Results

a. Although the number of fields within a message and the number of message types varies considerably among systems, a number of data types were identified as being useful in defining message formats and scenario data. The functional requirements were found to include accessibility to the data base by the three phases of test conduct: pre-test, real-time, and post-test.

b. Several commercial DBMS products were evaluated for integration with the TIS. The INGRES DBMS was chosen due to its VAX-11 compatibility, its relational view of data, and its additional forms, reports, and graphics capabilities.

c. A data base schema was developed that supports the representation of message formats, scenario data, and test configuration. This schema can be expanded to include various application-specific data types. A menu-driven interface was designed to support high-level access to the pre-test data base functions using the INGRES input forms software.

d. A procedure for defining message formats was developed. This procedure outlines the steps taken to decompose messages into fields, define field attributes and message structure, and define input forms for creating and modifying messages.

e. Character oriented message formats were constructed by applying the message format definition procedure to six MCS messages.

1.5 Analysis

The DBMS requirements imposed by the diversity of data characteristics and the evolving functional requirements associated with the development of scenarios for future systems were met through the use of a relational DBMS. An extensive software development effort would have been required to obtain the equivalent capabilities provided by the commercially available INGRES DBMS. The data base schema, high-level menu-driver, and procedures developed were validated using the character-oriented MCS as a test case.

1.6 Conclusions

The use of a commercially available DBMS was found to be a viable alternative for developing a data base of test messages. The relational model was found to be flexible enough to support character-oriented message formats. The data base schema and procedures developed during this investigation have provided useful input into the design of the pre-test component of the TIS.

1.7 Recommendations

The procedures and mechanism for defining character oriented message formats have been developed and validated for the TIS. Further investigation should be performed to enhance this capability to provide automated means of composing messages into events and scenarios for testing C^{3}I systems. A methodology investigation, Automated Aids to Test Data Generation, has been proposed to provide this capability.

2.0 DETAILS OF INVESTIGATION

This methodology investigation included four major tasks: the initial determination of the DBMS requirements for the test message data base, the selection and acquisition of a DBMS, the subsequent design and development of the supporting software, and the validation of this methodology via the utilization and maintenance of the message formats and scenario data.

2.1 Data Base Management System Requirements

DBMS requirements include data and functional requirements.

2.1.1 Data Requirements

The types of data required can be divided into five groups: message formats, scenario data, initialization data, and real-time data.

2.1.1.1 Message Format Data

a. The format of a message is the structure that allows specific controls or data to be identified by its position. This structure can be more conveniently decomposed into a sequence of fields, which possess the following attributes:

. Transmit Address - the byte and bit starting location within the message.

. Conversion Specification - the type of conversion required to change data from the data base format into the transmit format.

. Valid Range - the set of allowable data values.

. Default Value - the data value to be supplied through automated message generation.

b. The DBMS must not impose impractical limits on the numbers of fields and the lengths of messages, since these vary considerably from system to system.

2.1.1.2 Scenario Data

A scenario is a pre-test generated, time-sequenced collection of transmittable messages and non-transmittable commands that directs the operation of the test driver.

2.1.1.2.1 Transmittable Messages

a. The actual values of the data in some fields of a message may be of significance, either to the real-time handling or the post-test analysis. In this case, default data would suffice and in some messages, contiguous fields of this type could be combined into a single field.

b. On the other hand, the actual values of some fields are significant to the proper handling of protocols by the test driver. Fields of this type include:

- Bit patterns that frame the message or identify the format.
- Encryption sequences.
- Error detection and correction fields.

c. Fields with accountability information that are important to both real-time and post-test processing include:

- Source, relaying, and destination addresses.
- Retransmission and sequence numbers.
- Transmission time and priority.

2.1.1.2.2 Non-Transmittable Commands

Non-transmittable commands to the test driver include commands for:

- Initialization of hardware or software components of the driver.
- Initialization of driver controlling parameters (real-time variables).
- Synchronization.
- Insertion and deletion of transmittable messages.
- Modification of real-time variables.

2.1.1.3 Initialization Data

Initialization data describe the scenario-specific configurations of the hardware and software components of the test driver, and the values of some real-time variables.

2.1.1.4 Real-Time Data

Real-time variables that control time-critical processing have retrieval time requirements beyond current DBMS capabilities. However, the following types of real-time data have retrieval time requirements suitable to storage through a DBMS:

- Test configuration tables (network topology).
- Valid response tables.
- Store-and-forward data (message relaying).

2.1.2 Functional Requirements

The functional requirements of a test message data base can be divided into three groups: pre-test, real-time, and post-test functions.

2.1.2.1 Pre-Test Data Base Functions

Pre-test data base functions include message format definition and maintenance, scenario data generation and maintenance, and scenario file generation.

2.1.2.1.1 Message Format Definition and Maintenance

The test message data base should provide a user-friendly interface that supports the specification, insertion, deletion, modification, location, and output of the various message format components. In addition, the message formats that are generated need to be checked for completeness and consistency.

2.1.2.1.2 Scenario Data Generation and Maintenance

The test message data base must provide for the specification, insertion, deletion, modification, location, and output of message data and commands. Additional functions include the validation of user input data, the generation of default data, and the specification of errors.

2.1.2.1.3 Scenario File Generation

The scenario file stored in the test message data base will require restructuring into a time-sequenced scenario file in a format suitable for the test driver.

2.1.2.2 Real-Time Data Base Functions

The following real-time data base functions have been identified:

- Valid responses - the transmission of stored messages, triggered by messages received from the SUT.

- Message creation/injection - processing operator-entered messages.

2.1.2.3 Post-Test Data Base Functions

The message formats must be accessible to post-test data reduction routines.

2.2 Tradeoff Study and Acquisition of Data Base Management System

2.2.1 Tradeoff Study

Vendor supplied DBMS products were examined to determine the potential for meeting the data base requirements while reducing the development and maintenance costs associated with designing a DBMS. Of the eight commercial DBMS products surveyed, only two of them, INGRES and RAPPORT, appeared to offer a viable alternative to an "in-house" design. Both of these systems were compatible with the TIS hardware/software environment, included forms, statistics, graphics, and application development aids, and supported performance tuning. The INGRES features were found to be well integrated and flexible.

2.2.2 Acquisition of Data Base Management System

INGRES was selected and purchased because it had the following properties:

. Relational Model - INGRES relational data bases are easily modified and they support multiple user "views" of the data.

. VAX Compatibility - INGRES runs on the VAX-11 with built-in DECNET interface.

. High Level Language Interface - INGRES interfaces with FORTRAN, C, PASCAL, COBOL, and BASIC.

. Data Dictionary - INGRES system tables are available for use by application programs and for documenting the contents of the data base.

. Packages - INGRES software includes packages for designing user interfaces, input forms, reports, and graphics applications.

. Speed - Physical re-organization and indexing techniques are available for optimizing data operations.

. Capacity - INGRES data bases were found to have acceptable constraints with respect to the maximum number of tables (no practical limit), records within a table (no practical limit), fields within a record (127), and characters within a record (2000).

. Types of Data - INGRES supports integer, floating point, and character types of data, although it does not support binary data directly. Binary data must be converted into either a numeric or character representation before entering it into the DB.

. Default Values and Validations - INGRES supports default data values and various levels of data validation.

. Environment - INGRES supports a multiuser, interactive user environment.

. Bulk Loading - INGRES data bases can be bulk loaded and unloaded from VAX-11 files.

2.3 Software Design and Development

The ITIS, described in appendix C, was developed to automate the DT of the MCS. The ITIS has evolved into the TIS to meet additional interoperability testing requirements. This methodology investigation supported the design and development of the TIS's Test Message Data Base (TMDB) software.

2.3.1 Test Item Stimulator Design

a. The TIS has evolved from the ITIS in order to meet the additional requirements of interoperability testing. The TIS was designed to provide the capability to generate message traffic to support the testing of C^3I systems including:

. Joint Tactical Information Distribution System (JTIDS) class 2 terminal using TADIL-J messages.

. TSQ-73 host interface unit (HIU) using ATDL-1 and TADIL-B messages.

. Hawk HIU using ATDL-1 messages.

. Tactical computer terminal (TCT) HIU using MCS messages.

. TACFIRE Communications Control System (CCS) using TACFIRE CCS messages.

. Position Location Reporting System (PLRS)/JTIDS Hybrid (PJH) enhanced PLRS user unit (EPUU) using user read-out (URO) messages and EPUU messages.

b. To meet these requirements, the ITIS design was enhanced and incorporated into the TIS. The pre-test, real-time, and post-test functions run on the same VAX-11 based system. As a result, all three phases have access to the test message data base. The pre-test software provides the interactive capability to specify test conditions and maintain a library of message formats (MFL), and generate scenario events. Scenario messages are organized into events which together with the underlying format definitions comprise the Event Format Library (EFL). The real-time software generates, monitors, and records message traffic during the tests, providing for the interactive control of the test environment and parameters. The design also includes the capability for real-time operator-oriented event generation. The post-test software processes test data for test report generation. Figure 1 illustrates the data flow through the pre-test, real-time, and post-test functions.

2.3.2 Scenarios

The pre-test function supports on-line generation, review, and modification of test scenarios. The test scenarios consist of time-sequenced events and test actions which change the value of a system simulation parameter, provide test control information, or trigger required responses. Two types of events occur in scenarios:

a. Prescripted events - generated from pre-test operator-entered messages and directly transmitted to the SUT during real-time processing.

b. Real-time events - generated from pre-test operator-entered events that are manipulated during real-time testing to generate messages for transmission to the SUT.

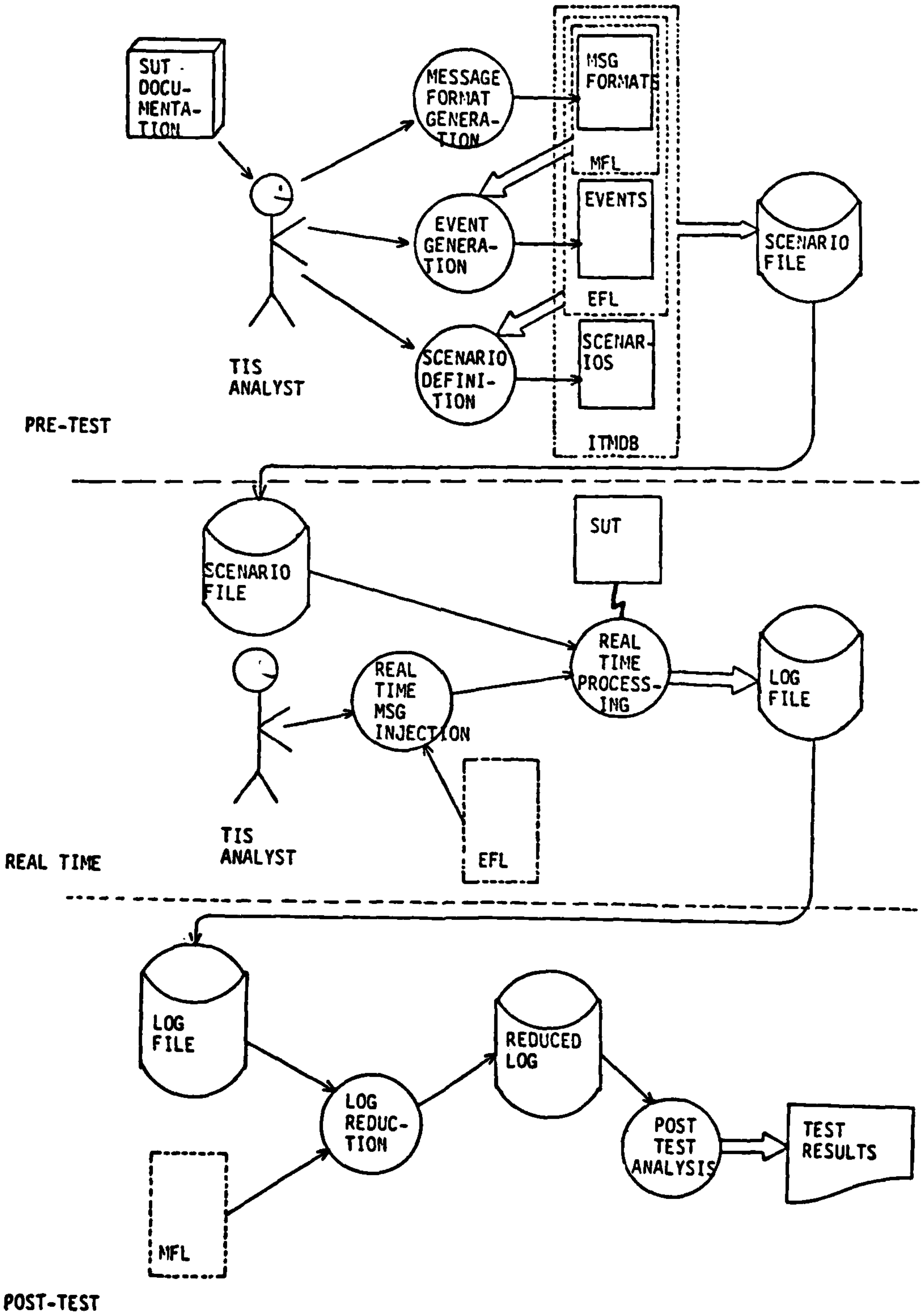


Figure 1. TIS Concept of Operations

2.3.3 Data Base Structure

The following data base schema has been developed for the TIS test message data base. This schema is illustrated in figure 2 (double-headed arrows are used to indicate one-to-many relationships). For discussion purposes, this schema is decomposed into four areas: test configuration data, scenario data, message formats, and application data.

2.3.3.1 Test Configuration Data

a. The TIS is designed to operate in a network configuration with other TISs during a test. Such a network is represented in the data base using the following types of records (component fields are listed, with key fields underlined):

. Network Description (Network ID, Network Description).

. Network Composition (Network ID, TIS ID, Test ID).

. Test Description (Test ID, Test Description, Date Created, Date Modified).

. Test Composition (Test ID, SSA, System ID, Scenario ID, Initialization File).

b. Each network is described by one Network Description record. Since a network may be composed of several TIS systems, one Network Composition record is used for each TIS, in each network.

c. Each TIS consists of four SSAs which are driven by separate scenarios and which are capable of stimulating different types of systems. The configuration of one TIS is described through the use of one Test Description record and one Test Composition record for each SSA. The System ID field of the Test Composition record is used to reference various records describing scenarios and formats. This convention allows for the independent naming of entities relating to a new system.

2.3.3.2 Scenario Data

a. The time-sequenced scenario files, used for real-time programming, are generated from the scenarios created and maintained through the pre-test function. These scenarios are represented in the data base by the following types of records:

. Scenario File (Scenario File, System ID, Scenario ID, File Creation Date).

. Scenario Description (System ID, Scenario ID, Scenario Description, Date Created, Date Modified).

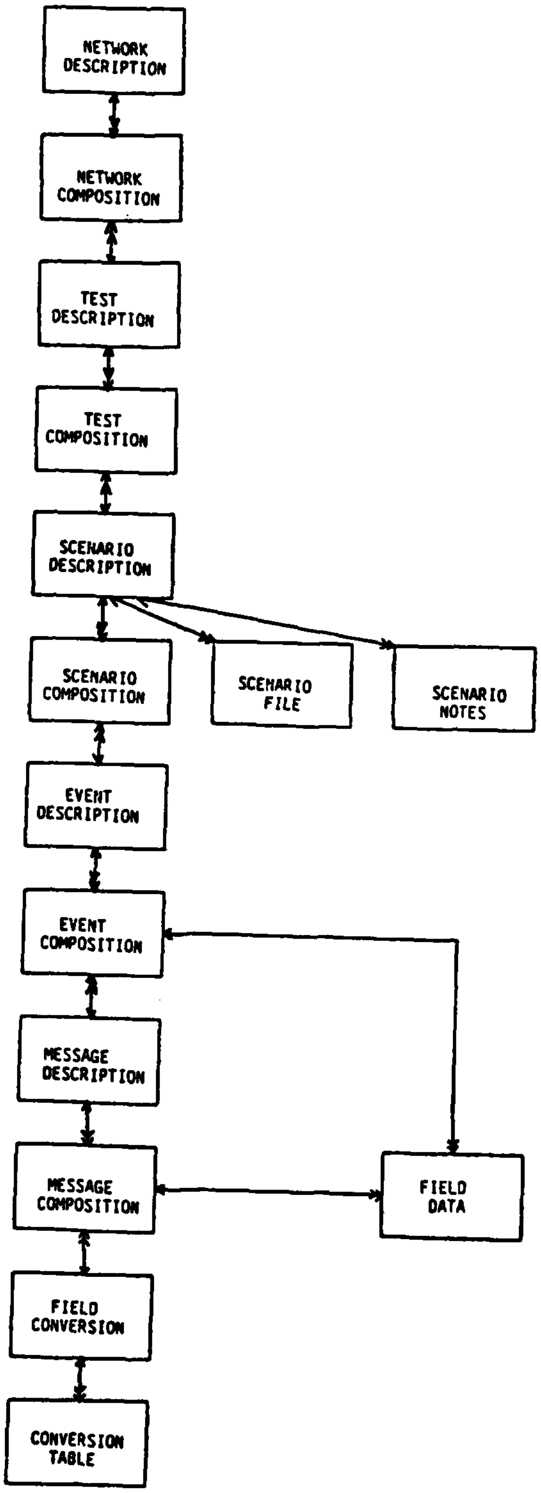


Figure 2. TIS Data Base Schema

- Scenario Notes (System ID, Scenario ID, Note #, Note Text).

- Scenario Composition (System ID, Scenario ID, Event ID, Event Start Time).

- Event Description (System ID, Event ID, Event Type, Event Description).

- Event Composition (System ID, Event ID, Message ID, Message Delta Time, Segment #, Source, Destination, Error, Response, Disposition, Message Index).

- Field Data (Message Index, Field #, Data).

b. A scenario file is associated with its data base representation through a scenario file record. Each scenario is described by one Scenario Description record, containing fixed format information, and any number of Scenario Notes records. Scenarios are composed of events represented by Scenario Composition records. Since a given event may occur several times within a scenario, particular instances of an event are distinguished through the Event Start Time field.

c. An Event Description record defines an event's type. This event type is used to indicate special processing for some events. Events are composed of messages, which are represented by Event Composition records. Each message has a delta time, which is offset from the event start time. An Event Composition record is uniquely described through its System ID, Event ID, Message ID, Message Delta Time, and Segment # fields. The Message ID field refers to the format of the message and the Segment # field allows for the definition of complex messages from simpler messages. The Message Index field is used as a space efficient key into Field Data records, replacing the above-mentioned five key fields.

d. Field Data records represent the actual message data. These records are indexed through their Message Index and Field # fields. For each message described by an Event Composition record, there will be one Field Data record for every field in the message.

2.3.3.3 Message Formats

a. A message format is the set of rules defining the properties and structure of a message. In addition to defining the structure of a transmitted message, the format defines the data base storage representation and the data entry rules, including the legal range of values and default values. The following types of records represent the message formats:

- Message Description (System ID, Message ID, Message Description, Form Name).

- Message Composition (System ID, Message ID, Field #, Form Field, Field ID, Field Location).

. Field Conversion (System ID, Field ID, DB Type, Conversion Type, Conversion Length, Field Description).

. Conversion Table (System ID, Field ID, ASCII, Bit Value).

Each type of message is represented by a Message Description record and one Message Composition record for each field of the message. The Form Name field of Message Description references the INGRES form which is to be used to enter and edit message data. This form contains the data entry rules for each field of a message. The Form Field of the Message Composition record refers to the corresponding set of rules which include the default value, validation check, validation error message, and other editing attributes. The Field ID field refers to the field type which may occur several times in a given message type or in more than one message type. The Field Location field refers to the byte and bit position of the field in the physically contiguous scenario file representation of this type of message.

b. The structure of a given type of field is defined through Field Conversion records. The DB Type field indicates the data base storage scheme, while the Conversion Type and Conversion Length fields indicate the data base-scenario-file conversion method. When indicated through the Conversion Type field, this conversion is represented by Conversion Table records, which associate operator-oriented character strings with binary values.

2.3.3.4 Field Access

a. The data base schema has, up to this point, treated all fields in a generic fashion. In some applications, such as automated message generation and post-test data reduction, a group of messages will contain a particular field of interest, yet having different formats, this field has different names and representations in each case. Such referencing can be handled in a general way using a record of the following type:

Field Access (Application ID, Application Field, System ID, Message ID, Field ID, Conversion)

b. Through the use of such records, specific field names and data base representations are independent of applications.

2.3.4 Data Base Functions

a. In addition to the usage of forms as a means of access to the data base records, the TIS pre-test software uses forms to create a high-level menu-driven interface accessing the data base functions. A detailed description of the menu driven interface appears in appendix D.

b. Figure 3 depicts the structure of the menu-driven interface to the TIS pre-test data base functions. These functions are organized at the highest level into five groups which are accessed through the following menus: Test Index, Scenario Index, Scenario File Generation, Scenario Preview Index, and EFL Menu. A detailed description of these functions appears in appendix D.

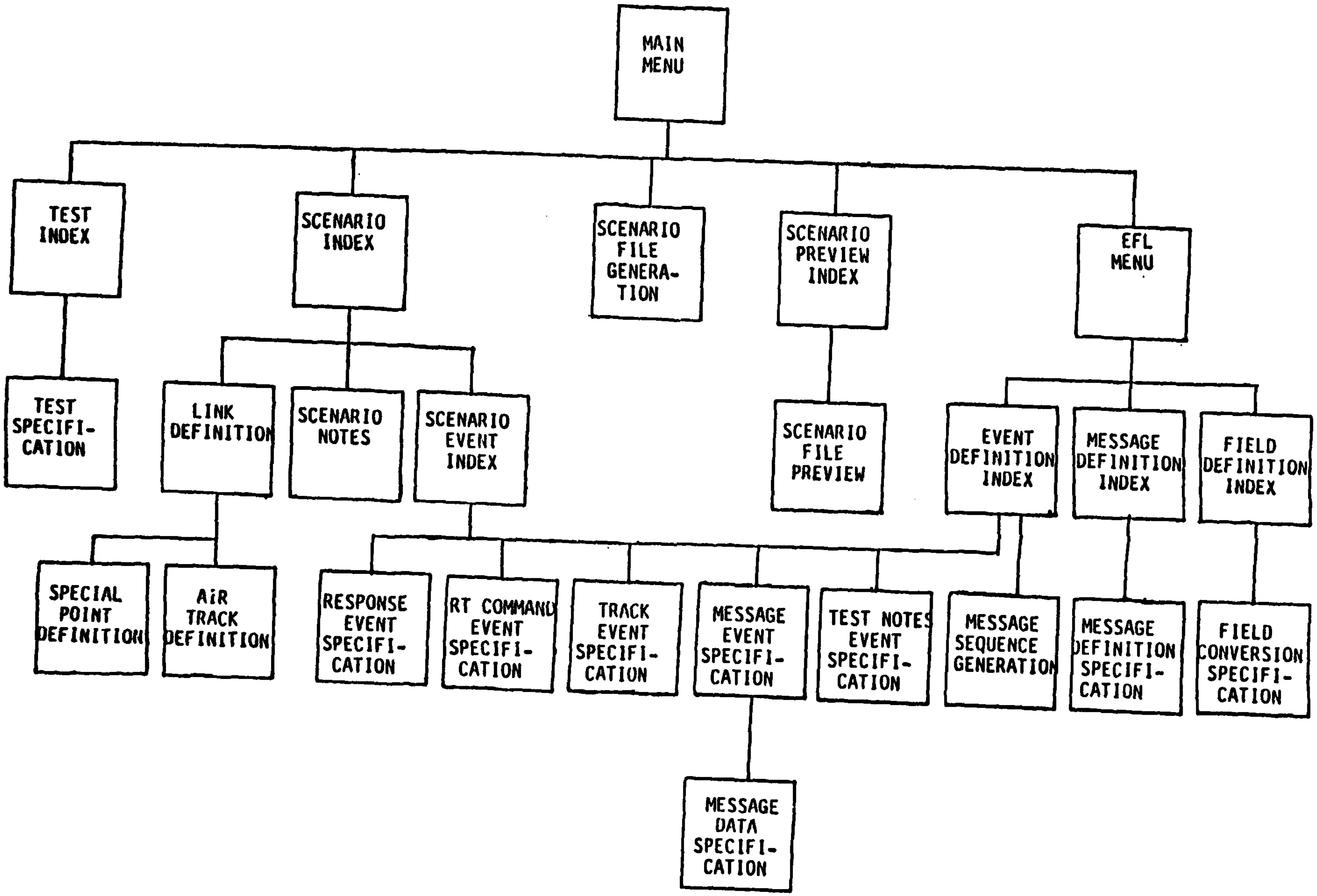

MAIN MENU
TEST INDEX
SCENARIO INDEX
SCENARIO FILE GENERA-TION
SCENARIO PREVIEW INDEX
EFL MENU
TEST SPECIFI-CATION
LINK DEFINITION
SCENARIO NOTES
SCENARIO EVENT INDEX
SCENARIO FILE PREVIEW
EVENT DEFINITION INDEX
MESSAGE DEFINITION INDEX
FIELD DEFINITION INDEX
SPECIAL POINT DEFINITION
AiR TRACK DEFINITION
RESPONSE EVENT SPECIFI-CATION
RT COMMAND EVENT SPECIFI-CATION
TRACK EVENT SPECIFI-CATION
MESSAGE EVENT SPECIFI-CATION
TEST NOTES EVENT SPECIFI-CATION
MESSAGE SEQUENCE GENERATION
MESSAGE DEFINITION SPECIFI-CATION
FIELD CONVERSION SPECIFI-CATION
MESSAGE DATA SPECIFI-CATION

Figure 3. Pre-Test Functional Block Diagram

2.4 Implementation of Message Format Libraries

The following sections describe the steps taken to define message formats for a new system.

2.4.1 Documentation Review

System documentation is reviewed to extract information related to the structure of messages. The message components are organized into two groups: those to be included in the scenario and those to be generated real-time by the SSA (e.g., error detection and correction (EDC) and framing information). Of those components to be included in the scenario, some may be modified during real-time processing.

2.4.2 Field Types

a. After examining all messages of one system, the field types are identified. Fields of a given type share the following attributes:

- Data Base Representation (data type).
- Conversion Method (data base representation to scenario representation).
- Field Length (scenario representation).
- Valid Range of Values.

b. Starting with an operational view of the field structures of the system's messages, fields can be combined or decomposed to simplify the format definition process. Fields that are indistinguishable with respect to testing may be combined, provided the increase in complexity is not inhibitive. Fields may be decomposed into smaller ones, simplifying the definition of conversion methods or validations. In addition, a complex field may be decomposed into fields having field types in common with other fields. Fields of a given type may occur within the same message or in different messages.

2.4.3 Message Structure

After defining the field types for a system, the individual message formats are defined in terms of their component fields. Each field in a message is defined in the following terms:

- Field Name (unique within a message).
- Field Type.
- Location (scenario representation).
- Default Value.

2.4.4 Forms

a. The first step in entering a message format into the data base involves the creation of an INGRES form. This form is accessed through the Message Event Specification menu during scenario generation. Each message format is associated with such a form, which is used to create and edit messages of that particular type.

b. These forms are generated in a two-pass process. In the first pass, the screen layout is defined. This definition includes the position of each field on the screen, the length and type of data, the tabbing sequence through the fields, and descriptive text on the screen about a particular field or the message in general.

c. During the second pass, the editing features dealing with each field are defined, one field at a time according to the tabbing sequence. The following information is defined for each field:

- Display Only Flag - when set, it causes the field to be skipped over during message editing.
- Default Value - initial field value.
- Field Name - the unique field name defined in section 2.4.3.
- Validation Rules - expressions defining constraints on the data accepted during editing.
- Validation Error Message - appears when data entered into the field is in disagreement with the validation rules.

2.4.5 Field Definition

The Field Conversion records required to define field types are entered through the Field Definition Index menu. These records are not redefined if they have been entered to define another message of the same system. The Field ID is the name of the field type. The DB Type, Conversion Type, and Conversion Length fields are supplied with the values defined in section 2.4.2. If the conversion method requires American Standard Code for Information Interchange (ASCII)-to-bit-value conversions, these conversions are entered through the Field Definition Specification menu, which is automatically accessed when the Field Conversion record is appended.

2.4.6 Message Definition

a. The actual entering of a message format to the data base starts with the definition of a Message Description record through the Message Definition Index menu. The message format is given a unique name within the system and is associated with the INGRES form defined in section 2.4.4 through the Form Name field. When the Message Description record is appended, one Message Composition record is generated for each field defined on the form. Since these records contain default Field Type and Field Address values, the Message Verification field of the Message Description record is set to indicate an incomplete format definition.

b. The Message Definition Specification menu, accessed through the specify command of the Message Definition Index menu, is used to update the Message Composition records. The Field ID, Field Type, and Field Address fields are completed using the definitions from section 2.4.3.

c. Once the Message Composition records reflect the message structure, the verify command of the Message Definition Index can be used to change the Message Verification field, allowing the message format to be used through the Message Event Specification menu.

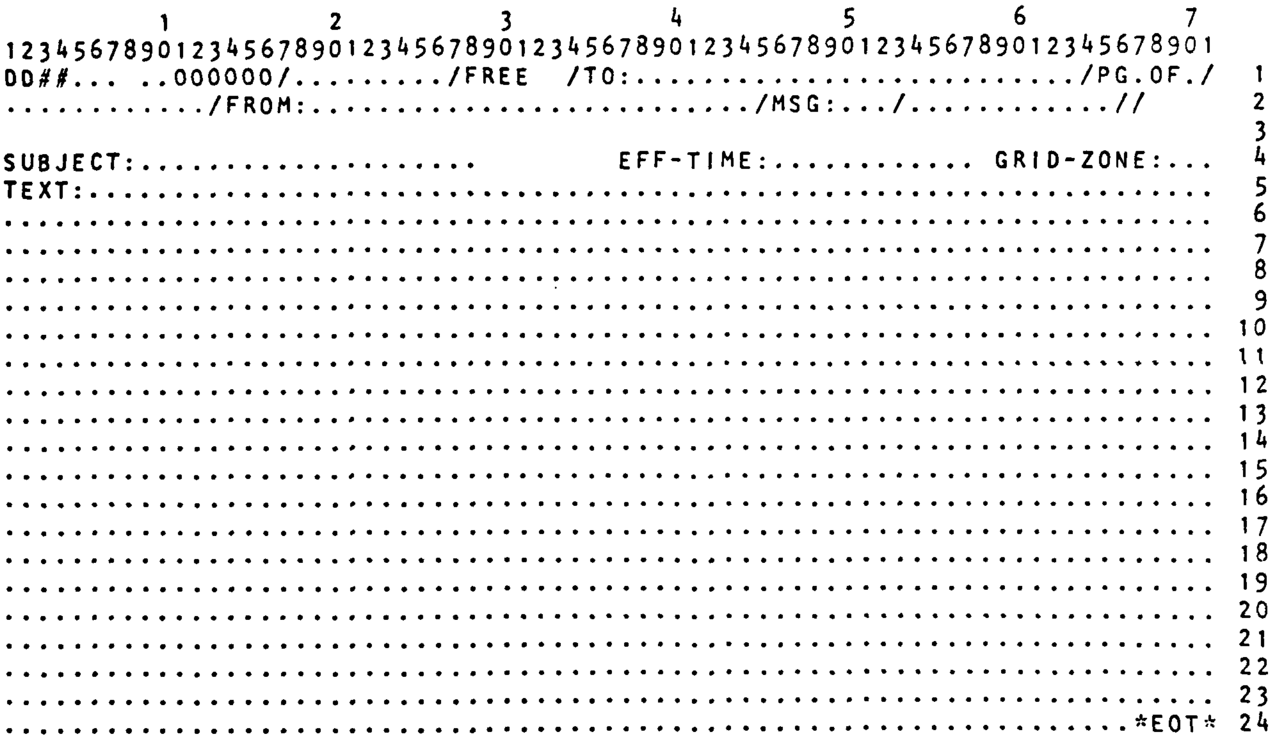
2.5 Maneuver Control System Specific Considerations

The following sections describe the steps taken to implement MCS message formats. A subset of the MCS messages was used to test the TIS pre-test software during its early stages of development. At that time, some of the pre-test software functions were not available and only a portion of the message formats were implemented. Six MCS messages were analyzed, their field types and message field structures were determined, and their editing forms were generated. The analysis of the field types was to have been followed by the definition of the Field Conversion records, just as the details of the message field structure and the forms were to have been the basis for the definition of Message Description and Message Composition records. However, the pre-test software supporting these definitions was not yet developed to a degree sufficient to actually store the definitions and create scenario data.

2.5.1 Message Analysis

a. The MCS messages were found to be character-oriented messages that are transmitted in 16 character blocks. Each transmitted message is preceded by four sync characters and terminated with a block containing at least four consecutive EOT characters. The MCS uses a 12/7 Hamming code for error detection and correction (EDC). This scheme allows the correction of single bit errors and the detection of double bit errors within a single character. In order to minimize the occurrence of multiple bit errors, the MCS uses a time dispersal coding (TDC) scheme, which regroups the bits within a block. The MCS can operate in a double block mode where each block is transmitted twice. It was determined that the scenario representation of a MCS message would be an unblocked ASCII character string. The EDC, TDC, and blocking would be controlled real-time, as would the generation of the leading sync and trailing EOT characters.

b. MCS messages are transmitted in two modes: unabridged and abridged. The MCS system uses plasma display terminals to display and edit messages. These messages appear in a 71 column by 24 line area of the screen. Figure 4 depicts such a message, with periods (.) representing the fields supplied by the operator. The remainder of the 1704 character area (not supplied by the operator) is the message skeleton. Messages transmitted in the unabridged mode are sent as they appear on the screen, i.e., as a 1704 character string. Abridged messages are transmitted without the message skeleton, having special field terminator characters separating the data fields. Furthermore, embedded blanks in the operator-entered fields are compressed and trailing blanks are suppressed, resulting in a much abbreviated transmitted message.



```
         1         2         3         4         5         6         7
12345678901234567890123456789012345678901234567890123456789012345678901
DD##... ..000000/........./FREE  /TO:........................../PG.OF./   1
............/FROM:........................../MSG:.../...........//        2
                                                                          3
SUBJECT:....................        EFF-TIME:............ GRID-ZONE:...   4
TEXT:..................................................................   5
.......................................................................   6
.......................................................................   7
.......................................................................   8
.......................................................................   9
.......................................................................  10
.......................................................................  11
.......................................................................  12
.......................................................................  13
.......................................................................  14
.......................................................................  15
.......................................................................  16
.......................................................................  17
.......................................................................  18
.......................................................................  19
.......................................................................  20
.......................................................................  21
.......................................................................  22
.......................................................................  23
..................................................................*EOT*  24
```

Figure 4. MCS FREE Message

c. The MCS graphics message is a bit-oriented message that is transmitted only in the abridged mode. Although bit-oriented messages are readily handled by the TIS pre-test software, a mechanism has not been developed for entering graphics-oriented data into the data base.

2.5.2 Message Structure and Field Types

a. The messages of the MCS were analyzed to determine the field structure of the messages and the characteristics of the fields. Three ways of representing the abridged and unabridged forms of the messages were identified:

. Abridged and unabridged forms of a message were to be considered separate messages.

. A special flag would direct the conversion of messages from a common data base representation into the appropriate scenario file representation.

. Only abridged messages would be included in the scenario file; the abridging would be handled by real-time processing.

Of these methods, defining separate formats was the method most compatible with the early TIS design.

b. The abridged and unabridged structures were determined for the following MCS messages: Commanders Report (CDRREP), Clear Report (CL-RPT), Down Wind (DN-WND), Free Text (FREE), Spot Report (SPOT), and Unit Task Organization (UTO). These messages were decomposed into a combined total of approximately 1200 message fields, which were grouped into over 50 field types. Table I details the structure of the unabridged FREE message. The name, type, and location columns correspond to the Form Field, Field ID, and Field Location fields of the Message Composition records. The default value column corresponds to the data used to define the INGRES form used to enter and edit FREE messages. Table II lists the field types used to define the format of the unabridged FREE message. The type and length columns correspond to the Field ID and Conversion Length fields of the Field Conversion records. The valid ranges are used to define restrictions on input to each field of the FREE message form. The name and type identifiers used in these tables were arbitrarily chosen to reflect the field useage.

The message structure for the abridged FREE message does not contain the skeleton fields, but instead it contains terminator fields to separate the data fields. The field types of the abridged data fields are the same as the unabridged types, except that blank compression and suppression are specified by the conversion method.

2.5.3 Form Definition

a. INGRES forms were constructed for the six unabridged messages. The first step taken in creating each form was the definition of the size of each field and position on the screen. The forms were defined using the 71 column by 24 line layout seen on the MCS plasma display terminals. After defining the screen layout, the following form attributes were defined on a field-by-field basis:

Table I

Message Composition for the
MCS Unabridged FREE Message

NAME	TYPE	LOCATION	DEFAULT VALUE
HDR0	D2	0	DD
HDR2	NODE	2	01
HDR4	D1	4	U
HDR5	ETE	5	N
HDR6	REC	6	(blank)
HDR7	D1	7	(blank)
HDR8	ADDR	8	00
HDR10	NODE	10	01
HDR12	RTN	12	0
HDR13	SEQ	13	000
D11	D1	16	/
PRECED	PRE	17	ROUTINE
D13	D11	26	(blank)
DESTID	C26	37	(blank)
D15	D3	63	/PG
PAGE	PG	66	1
D17	D2	67	OF
PAGES	PG	69	1
D19	D1	70	/
CLASS	CLASS	71	UNCLASSIFIED
D21	D6	83	/FROM:
ORIGID	C26	89	(blank)
D23	D5	115	/MSG:
SEQ	SEQ	120	000
D25	D1	123	/
XTIME	DTG	124	051345ZFEB80
D27	D6	136	//
D28	D71	142	(blank)
D29	D8	213	SUBJECT:
SUBJ	C20	221	(blank)
D31	D17	241	EFF-TIME:
EFFT	DTG	258	051345ZFEB80
D33	D11	270	GRID-ZONE
GZONE	GZ	281	32U
D35	D5	284	TEXT:
TXT1	C66	289	(blank)
TXT2	C71	355	(blank)
TXT3	C71	426	(blank)
...	...	...	...
TXT19	C71	1562	(blank)
TXT20	C66	1633	(blank)
D56	D5	1699	*EQT*

Table II

Field Types Associated with the
MCS Unabridged FREE Message

TYPE	LENGTH	VALID RANGE
D*	*	no lower case letters, display only
NODE	2	01-99
ETE	1	Y,N
RFC	1	Blank, 1-8
ADDR	2	00-99
RTN	1	0-9
SEQ	3	000-999
PRE	9	ROUTINE, PRIORITY, IMMEDIATE, FLASH
C*	*	no lower case letters
PG	1	1-9
CLASS	12	UNCLASSIFIED, CONFIDENTIAL, NATO CONF,...
DTG	12	010000AJAN00 - 312359ZDEC99
GZ	3	00A - 99Z

(1) Display only flag. This flag was set for each of the message skeleton fields to prevent the operator modification of these fields.

(2) Default value. This value is used to initialize the fields during message creation. Both skeleton fields and data fields have default values.

(3) Field name. This name is the same used to define the Form Field field of Message Composition records.

(4) Validation expression. This attribute defines the range of values that can be entered into a field.

(5) Error Message. This message is displayed to the operator when an illegal value is entered. Neither the validation expression nor the error message are defined for message skeleton fields.

After defining the field attributes, the forms were named and stored in the data base.

b. The following modifications can be made for defining forms for abridged messages:

. Trim fields can be used to display the message skeleton without affecting the format definition.

. Field terminators can be defined as display only fields that are located outside the 71 column by 24 line message area.

APPENDIX A

METHODOLOGY INVESTIGATION PROPOSAL

January 1983

METHODOLOGY INVESTIGATION PROPOSAL

1. TITLE. Interoperability Test Message Data Base.

2. CATEGORY.

a. Thrust Areas

(1) VISTA

(2) DC^{3}I

b. Sub-Areas

(1) Interoperability

(2) Software

3. INSTALLATION. US Army Electronic Proving Ground, Fort Huachuca, Arizona 85613.

4. PRINCIPAL INVESTIGATOR. Leslie F. Claudio, Software and Automation Branch, STEEP-MT-DA, AUTOVON 879-1879.

5. STATEMENT OF THE PROBLEM. A rapidly increasing number of automated communications-electronics (C-E) systems are being developed for use by the field army. The comprehensive testing of the interoperability of these complex systems is critical to their performance evaluation. Interoperability testing is accomplished by exposing the system under test to a set of input messages which are representative of outputs from other systems and observing the response of the system under test.

a. As interoperable systems undergo independent evolution, that set of messages which represents valid system interoperation will change. System level testing of message driven systems requires a "library" of currently valid input messages for application as stimuli and output messages for system performance determination. A library defining legal or valid message does not exist within TECOM.

b. If a library could be developed, a means must also be established to maintain the library. TECOM does not have a means of maintaining the library.

6. BACKGROUND.

a. History. Army and DOD have developed and are continuing to develop a number of major, automated, Command, Control, Communications and Intelligence (C^{3}I) Systems. These include TACFIRE, TOS, TACELIS, AGTELIS, AN/TYC-39, AN/TSQ-73, ASAS, GOTAS, and REMBASS. These systems are designed to interface with a large number of systems and to operate in a highly interactive environment. The critical element in the success or failure of these C^{3}I systems will be their interoperability and performance under load in a tactical environment. A capability is being developed, the Interim Test Item Stimulator (ITIS), which can fully evaluate the interoperability of these complex data handling systems.

Interoperability Test Message Data Base (cont)

b. Progress. This investigation was partially funded and initiated in January 1980. A commercially available DBMS was not found at project initiation which could see EPG requirements. A basic capability was developed and demonstrated as part of the ITIS pre-test system.

7. GOAL. To provide a method for developing, using, and maintaining an automated data base of digital stimulus messages and valid response messages between each system and all other systems with which it must interoperate for both technically and tactically significant scenarios.

8. DESCRIPTION OF INVESTIGATION.

a. This investigation will develop a methodology for an automated data base from which a test officer may select a set of test messages (and anticipated responses) appropriate of the interface(s) being tested which reflect the current revision levels of the software in a system under test and the various systems with which it must interoperate. The task will result in the design of the data base, the initial filling of it, and the establishment of procedures for using and maintaining it.

b. USAEPG will conduct the investigation in five phases:

(1) The technical requirements for the data base system will be determined. The technical requirements will consider the types of data which must be included in the data base, the quantity of data, the structure of the data base which minimizes cost and retrieval time, search requirements, the need to allow access from other systems, and the need for interactive operation, security requirements, and update facility.

(2) The set of operations which must be performed on the data base by a test officer will be determined. The set will be designed to make efficient use of his time but limit his opportunity to introduce errors in the design of his test cases. He will use the operations to select the messages from the data base, alter them where necessary, and assemble them for output on removable media. The removable media will be used as an input media by the test driver system.

(3) The data base system will be selected and purchased or designed and implemented. Procedures for using and maintaining the data will be developed. The data base software will be run on then existing computer resources.

(4) The data base will be initiated with a set of test messages for Maneuver Control System and TACFIRE that is representative of the messages likely to be desired by a test officer for a live test of a real system. These messages will be either composed manually or taken from existing files.

(5) Operation of the data base, use procedures, and all documentation will be validated.

Interoperability Test Message Data Base (cont)

c. USAEPG will conduct the investigation as follows:

MILESTONE/PHASE	SCHEDULE							
	FY 83 (Qtrs)				FY 84 (Qtrs)			
	1	2	3	4	1	2	3	4
Requirements Phases		X	X					
Implementation			X	X	X	X	X	
Data Base Development				X	X	X	X	
Validation				X			X	X
Reporting		X		X		X		X

d. This investigation will result in a new procedure for assembling large sets of digital test messages. The procedure will allow utilization of automatic test drivers as system test instrumentation.

e. Environment Impact Statement. The execution of the investigation will not have an adverse impact on the quality of the environment.

f. Health Hazard Statement. The execution of the investigation will not involve health hazards to personnel.

9. JUSTIFICATION.

a. Mission and Impact Statement

(1) Association with Mission. EPG's primary mission is to conduct development testing of C-E equipment and systems. In support of this mission, EPG has developed extensive experience in compatibility vulnerability, electronic warfare, and intelligence testing, and more recently pioneered efforts in automated system (software) testing. Interoperability testing represents the integration of these individual test responsibilities.

(2) Capability, Limitations, Improvements and Impacts. The present capability consists of manually prepared preformatted tapes and manually prepared scripts. It cannot accommodate large volumes of messages, cannot respond rapidly to test requirements, cannot respond rapidly to changes in test item software, and is very sensitive to the introduction of errors by humans. The improvements that will be gained by this investigation include: a reduction of errors contained in test messages, coordination of test messages with other test messages and the data bases in the systems under test, and management of large volumes of test messages. Media will be compatible with automated test drivers, and the assembly of test message will be responsive to changes in the test and changes in the test software. Failure to complete this investigation will prevent the utilization of automated test drivers as input devices for testing message--driven systems. Interoperability and load testing of such systems will be impossible.

Interoperability Test Message Data Base (cont)

b. Workload. The following major field army automated systems are currently under development and are programmed for testing or retesting.

	TECOM			FY			
System	Priority	83	84	85	86	87	88
RPV			DT-II				
PJH				DT-II	DT-II		
JTIDS			DT-II				
SHORAD						DT-II	
ATHS		DT-II					
IINS		DT-II					
AFATDS							DT-I
ASAS				DT-I		DT-II	

c. Dollar Savings. In addition to providing a capability which was previously not available within TECOM, the manhour savings as gleaned from actual use of the initial capability on MCS is estimated to be a factor of 30:1.

d. Association with Requirements Documentation. The Army Battlefield Automation Interoperability System Engineering Management Plan (BAISEMP), dated November 1978, outlines the requirements for interoperability testing.

e. Other. None.

10. RESOURCES.

a. Financial

(1) Funding Breakdown

	Dollars (Thousands) FY 83	
	In-House	Out-of-House
Personnel Compensation	15.0	
Travel	6.0	
Contractual Support		44.0
Consultants & Other Services		10.0
Material & Supplies	5.0	
Equipment		
ADP	10.0	
Subtotals	36.0	54.0
FY Totals	90.0	

Interoperability Test Message Data Base (cont)

(2) Explanation of Cost Categories

(a) Personnel Compensation. Direct labor, one civilian part time chargeable to the investigation.

(b) Contract Support. Required to complement in-house resources -- includes consultants, materials, etc.

b. Anticipated Delays. None.

c. Obligation Plan. (Fiscal quarters FY 84)

	FQ	1	2	3	4	TOTAL
Obligation Rate (Thousands)		60	10	10	10	90

d. In-House Personnel

	FY 83	
	Manhours	
	Required	Available
Elec Engr GS-0855	300	300
Comp Spec GS-0334	300	300

Resolution of Non-Available Personnel. Required personnel are expected to be available.

11. INVESTIGATION SCHEDULE.

	FY 83												FY 84											
	O	N	D	J	F	M	A	M	J	J	A	S	O	N	D	J	F	M	A	M	J	J	A	S
In-House	-	-	-						-	-	-	R						-	-	-	-	-	-	R
	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.		
Contract	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-		

Symbols:		
	- - -	Active investigation
	. . .	Contract monitoring
	R	Final report due at HQ, TECOM

12. ASSOCIATION WITH TOP PROGRAM.

a. No TOPs will be revised as presently planned.

b. Initial plan does not include any TOP but the investigation will supply the decision.

FOR THE COMMANDER:

(signed)

MELVIN FOWLER
LTC, Sig C
Director of Materiel Test

APPENDIX B

ACRONYMS

ACSII	American Standard Code for Information Interchange
ATDL-1	Automated Tactical Data Link (message format used by TSQ-73 and I-HAWK)
BRTS	Basic Real-Time System
CCS	Communications Control System (Advanced Field Artillery Tactical Data Systems)
C^3I	Command, Control, Communications, and Intelligence
DB	Data Base
DBMS	Data Base Management System
DOD	Department of Defense
DT	Developmental Testing
EDC	Error Detection and Correction
EFL	Event Format Library
EPUU	Enhanced PLRS User Unit
HIU	Host Interface Unit
INGRES	Interactive Graphics and Retrieval System
ITIS	Interim Test Item Stimulator
ITMDB	Interoperability Test Message Data Base
JTIDS	Joint Tactical Information Distribution System
MCS	Maneuver Control System
MFL	Message Format Library
MFLG	Message Format Library Generator
MLF	Message Log File
MMFL	Master Message Format Library
MSDB	Message Scenario Data Base
MSDBG	Message Scenario Data Base Generator
MSF	Message Scenario File
MST	Message Scenario Tape
MSTG	Message Scenario Tape Generator

PLRS	Position Location Reporting System
PJH	Position Location Reporting System/JTIDS Hybrid
PTAS	Post-Test Analysis System
SSA	System-Specific Applique
SUT	System Under Test
SYNC	Synchronization
TACFIRE	Tactical Fire Direction System
TCT	Tactical Computer Terminal
TDC	Time Dispersal Coding
TIS	Test Item Stimulator
TMDB	Test Message Data Base
URO	User Read-Out
USAEPG	U.S. Army Electronic Proving Ground

APPENDIX C

INTERIM TEST ITEM STIMULATOR DESIGN

1.0 Interim Test Item Stimulator Design

a. The ITIS consists of six programs: the Message Format Library Generator (MFLG) program, the Message Scenario Data Base Generator (MSDBG) program, the Message Scenario Tape Generator (MSTG) program, the Basic Real-Time System (BRTS), the System-Specific Applique (SSA), and the Post-Test Analysis System (PTAS). The first three of these programs run on a Data General ROLM 1666 and constitute the pre-test system of ITIS. The BRTS and SSA programs run on a Data General Eclipse S140, forming the real-time system; and the PTAS program, running on a Digital Equipment VAX 11-780, represented the post-test system. The data/control flow of these programs is shown in figure C-1.

b. The MFLG program is a multi-pass batch program which generates the Message Format Library (MFL) files that contain system-specific information on how to construct and alter a message for transmission to the SUT. The Master Message Format Library (MMFL) file defines the structure of the MFL files.

c. The MSDBG program is an interactive program that builds and edits messages for use in the real-time testing of a SUT. These messages and relevant control data are stored in a Message Scenario Data Base (MSDB) file. Using the MFL files, this program guides the operator through correct message construction and editing, retrieval and editing of completed messages from the data base, and the creation of invalid messages for system error-detection testing and auto-duplication of messages for system throughput testing.

d. The MSTG program is a batch program which generates a Message Scenario Tape (MST) by merging up to nine MSDBs. The program allows for the definition of a test configuration and the specification of message selection and deletion criteria. The selected messages are prepared for transmission by the ITIS to the SUT, and sequenced on the MST according to each messages' requested transmission time.

e. A tape-to-disk program reads the ROLM MST into a Data General Eclipse compatible Message Scenario File (MSF).

f. The real-time testing is accomplished through the use of the BRTS program coupled to an SSA. The SSA performs the protocol processing and all other SUT-specific processing. The BRTS program handles those functions that are common to all tests. These functions include reading the MSF, transmitting the information to the SUT, and recording all stimuli and responses on the Message Log File (MLF).

g. A disk-to-tape program outputs the MLF onto a VAX-11 readable log tape.

h. The PTAS program is a library of analysis and reporting procedures to analyze a test's scenario and log tapes, generating reports on the performance of the SUT.

i. The ITIS has a limited capability as a test driver having been designed to test one system at a time. The ITIS scenarios are generated from a single MFL, which can contain up to 50 character-oriented message formats.

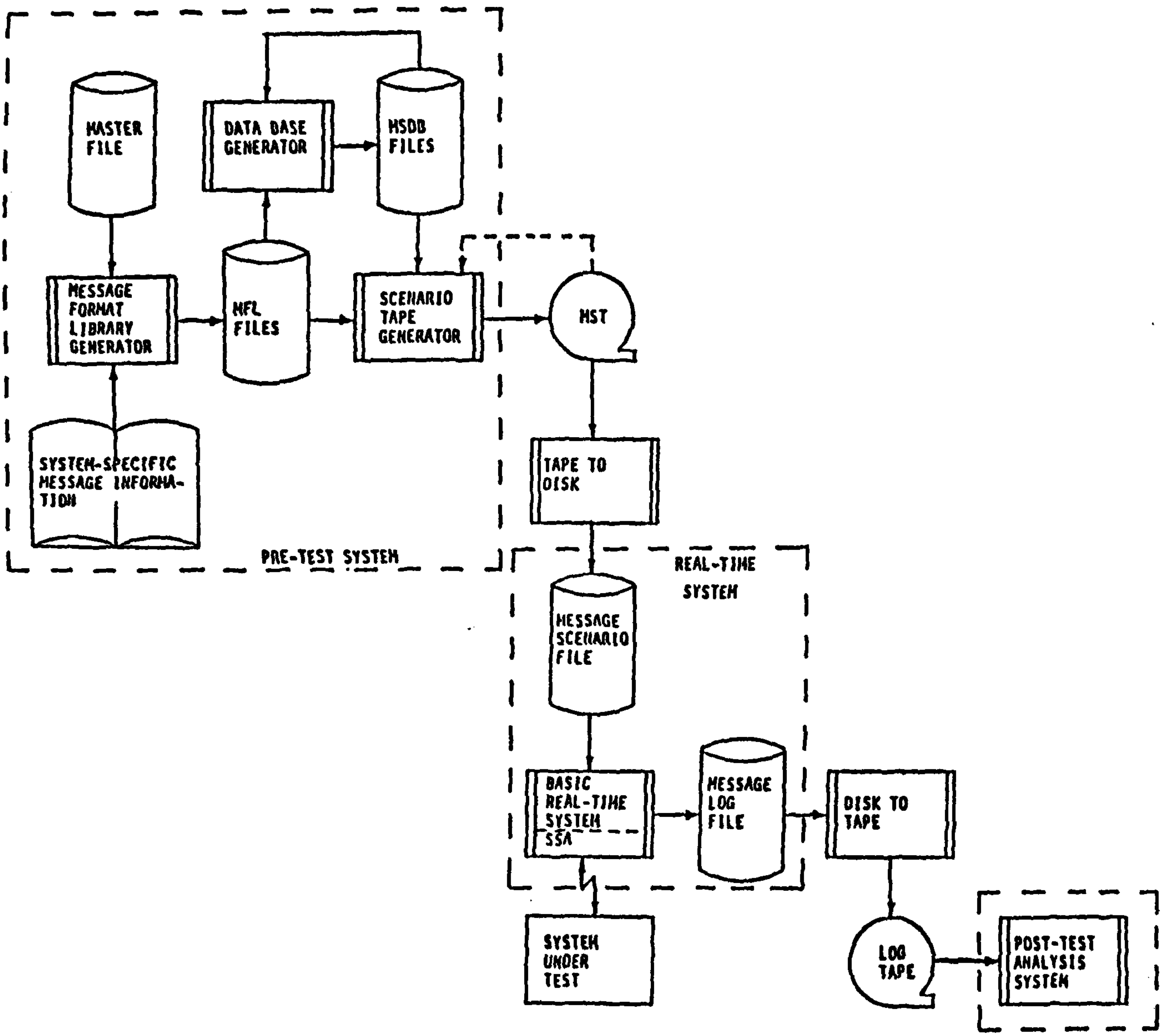


Figure C-1. Overview of ITIS Program Interaction

APPENDIX D

DATA BASE FUNCTIONS

DATA BASE FUNCTIONS

1.0 Typical Menu

TIS pre-test software uses forms to create a high-level menu-driven interface accessing the data base functions. Figure D-1 illustrates such a menu screen. The following menu commands are typical of the TIS pre-test menus:

a. Append. This command attempts to add the data in the edit buffer into the table. If the data entered violates any integrity rule, the append fails and a message appears in the warning/prompt line. Otherwise, the data is inserted into the table according to the sorting rules, causing the table to scroll until the cursor is positioned over the row just added.

b. Replace. This command attempts to replace the row under the cursor with the edit buffer data. If any integrity rule is violated, the replacement fails and a warning message is displayed.

c. Delete. This command attempts to delete the row under the cursor. A warning appears if the deletion is not allowed for a particular row.

d. Edit. This command copies data from the row under the cursor to the edit buffer, where it can be modified.

e. Copy. This command generates a new row, based on the row under the cursor.

f. Specify. The menu for accessing the next lower level of commands is reached through this command.

g. Find. This command scrolls the table, positioning the cursor over the appropriate row.

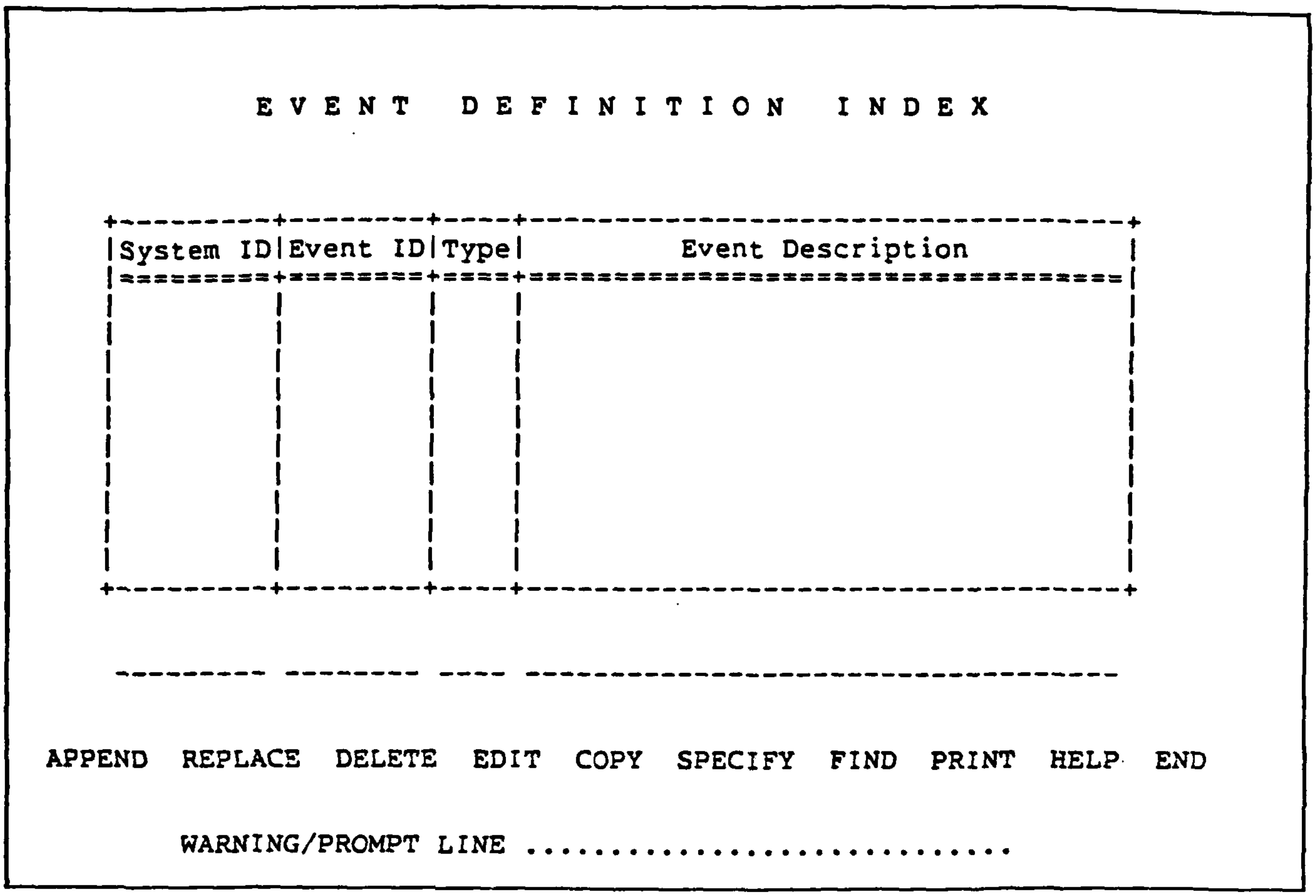
h. Print. A hard copy of the entire table is generated through this command.

i. Help. This command displays one or more help screens. The menu is then restored to its status prior to the help command.

j. End. This command returns the operator to the menu through which this menu was specified.

2.0 Menu-Driven Interface

The TIS pre-test data base functions are organized at the highest level into five groups which are accessed through the following menus: Test Index, Scenario Index, Scenario File Generation, Scenario Preview Index, and EFL Menu. These functions are described in the following sections.



```
               E V E N T   D E F I N I T I O N   I N D E X


   +---------+--------+----+-----------------------------------+
   |System ID|Event ID|Type|         Event Description         |
   |=========+========+====+===================================|
   |         |        |    |                                   |
   |         |        |    |                                   |
   |         |        |    |                                   |
   |         |        |    |                                   |
   |         |        |    |                                   |
   |         |        |    |                                   |
   |         |        |    |                                   |
   |         |        |    |                                   |
   |         |        |    |                                   |
   |         |        |    |                                   |
   +---------+--------+----+-----------------------------------+


    --------- -------- ---- -----------------------------------

APPEND  REPLACE  DELETE  EDIT  COPY  SPECIFY  FIND  PRINT  HELP  END

        WARNING/PROMPT LINE .............................
```

Figure D-1. Typical Pre-Test Menu

2.1 Test Index

a. The Test Index menu provides access to the data base functions dealing with the organization of tests. The following test description record fields are operated on through this menu:

- Test ID.
- Test Description.
- Date Created.
- Date Modified.

b. The following command details relate to the Test Index menu:

(1) Append. This command fails if a Test Description record having the same Test ID exists.

(2) Replace. This command fails if the Test ID is modified.

(3) Specify. This command provides access to the Test Specification menu.

c. The Test Specification menu operates on the following Test Composition fields:

- SSA.
- System ID.
- Scenario ID.

d. The following details apply to the Test Specification menu commands:

(1) Append. This command fails if another Test Composition record for this test has the same SSA value.

(2) Replace. This command fails if the SSA field has been modified.

2.2 Scenario Index

a. The Scenario Index menu provides access to the data base functions operating on Scenario Description records. The following fields are visible to the operator:

- System ID.
- Scenario ID.
- Scenario Description.
- Date Created.
- Date Modified.

b. The following details apply to the Scenario Index menu commands:

(1) Append. This command fails if another Scenario Description record exists with the same System and Scenario ID.

(2) Replace. This command fails if the System ID or Scenario ID fields are modified, if the scenario has been used to define a test, or if a scenario file has been generated from it.

(3) Delete. This command fails if a scenario file has been generated for the scenario or if the scenario is used to define a test. The associated Scenario Notes and Scenario Composition records are also deleted.

(4) Copy. This command generates a new Scenario Description record and its associated Scenario Notes and Scenario Composition Records.

(5) Find. This command locates a scenario based on its System ID and Scenario ID.

(6) Specify. This command accesses the data base functions operating on Scenario Composition records through the Scenario Event Index menu.

(7) Notes. This command accesses the data base functions operating on Scenario Notes records.

c. The Scenario Event Index commands operate on these Scenario Composition fields:

. Start Time.

. Event ID.

d. In addition, the following Event Description fields are displayed for each scenario element:

. Event Type.

. Event Description.

e. The following details apply to the Scenario Event Index menu:

(1) Append. This command fails if the Start Time is not unique, if the Event ID is not valid, if the scenario has been used to define a test, or if a scenario file has been generated from this scenario.

(2) Replace. This command fails if the Start Time has been modified, if this scenario has been used to generate a scenario file, or if this scenario has been used to define a test.

(3) Delete. This command deletes the specified event. The delete fails if this scenario has been used for scenario file generation or test specification.

(4) Find. This command locates an event based on its Start Time.

(5) Examine. This command allows access to the Event Composition data associated with the specified event. The appropriate menu is displayed depending on the event type.

f. These menus are the same ones accessed through the specify command of the Event Definition Index, although only the display functions are available.

2.3 Scenario Preview and Scenario File Generation

a. The Scenario Preview and Scenario File Generation menus display the following Scenario Description record fields:

. System ID.

. Scenario ID.

. Scenario Description.

. Date Created.

. Date Modified.

b. Both menus locate a given scenario through its System ID and Scenario ID fields with a find command. The Preview and Scenario File Generation functions construct time-sequenced files from the relational representation of the scenario. In the case of the preview function, the output is in a hardcopy or terminal display format. In addition, the preview function can operate on a subset of the scenario, based on operator-input parameters. The Scenario ID field of the specified record indicates which Scenario Composition records are to be used.

c. The scenario is processed one field at a time, using data from the following types of records:

(1) Scenario Composition. The Event Start Time field is used to compute the scenario time for each message. The System ID and Event ID fields specify the Event Composition records associated with each event.

(2) Event Composition. The Message Delta Time field is used in conjunction with the Event Start Time to compute the scenario time for each message of an event. The System ID and Message ID fields specify the Message Composition records associated with each message. The Message Index field is used to access the Field Data records associated with each message.

(3) Message Composition. The System ID and Message ID fields specify the Field Conversion record associated with each field. The Field # field is used in conjunction with the Message Index to access the data of a particular field. The Field Location field indicates the byte and bit position of the field within the message.

(4) Field Conversion. The Conversion Type and Conversion Length fields indicate the method to be used when converting the data base representation of a field into its scenario file format. If indicated by the conversion type, the System ID and Field ID fields specify the Conversion Table records associated with the field.

(5) Field Data. This record contains the scenario data in its data base representation. For fields with conversions using Conversion Table records, the particular Conversion Table record is specified by the ASCII value of this data.

(6) Conversion Table. The Bit Value field contains the specific bit pattern to be used in the scenario file.

2.4 Event Format Library Menu

The Event Format Library (EFL) Menu provides access to the data base functions dealing with the Event Format Library EFL. Fields that are defined are used for defining messages, which are used to define events in the library. Events are, in turn, used for defining scenarios. The Event Definition Index, Message Definition Index, and Field Definition Index menus are accessed from the EFL Menu.

2.4.1 Event Definition Index

a. The functions operating on Event Description records are accessed through the Event Definition Index, which is illustrated in figure 2-3. The following Event Description fields are affected:

- System ID.

- Event ID.

- Event Type.

- Event Description.

b. The following command details are specific to this menu:

(1) Append. This command fails if an Event Description record having the same System ID and Event ID exists, or if the Event Type is invalid.

(2) Replace. This command fails if the System ID is changed or if the Event ID is changed and one or more scenarios have been defined using it.

(3) Delete. In addition to deleting the specified Event Description record, this command deletes all Event Composition records associated with it, as well as all field data records associated with the deleted Event Composition records. This command fails if this event has been used to define any scenario.

(4) Copy. This command generates new Event Composition and Field Data records as well as a new Event Description record. In addition, if specified, the Message Delta Time fields can be adjusted by an operator-entered offset time.

(5) Specify. This command accesses the appropriate menu, based on the Event Type field of the specified record. The data base functions operating on the Event Composition records are implemented using application-specific combinations of input forms and commands. Although all events have the same data base structure, distinct event types are seen by the pre-test operator,

including message events, real-time commands, test notes, and response events. Although all events are built from messages, which are built from fields, the application-specific software presents a view of only the necessary subset of data and commands.

(6) Find. This command locates a record based on its System ID and Event ID fields.

2.4.2 Message Event Specification

a. The most general view of event composition is seen through the Message Event Specification menu, which operates on the following Event Composition fields:

. Message ID.

. Message Delta Time.

. Source.

. Destination.

. Response.

. Errors.

. Disposition.

b. The following command details are specific to the Event Message Specification menu:

(1) Append. This command fails if the Message ID field does not reference a valid message format. An Event Composition record cannot be appended, replaced, or deleted if that event has been included in any scenario.

(2) Replace. This command fails if the Message ID is modified.

(3) Delete. This command deletes the associated Field Data records as well as the specified Event Composition record.

(4) Copy. This command generates a new Event Composition record with an operator-entered Message Delta Time field, along with new Field Data records.

(5) Specify. This command causes the appropriate INGRES form to be displayed for editing the actual message data. These INGRES forms are defined for a specific message format and are accessed by name through the Form Name field of the Message Description record, accessed through the Message ID field of the specified Event Composition record. The message data is fetched for editing from Field Data records indexed through their Message Index and Field # fields.

(6) Find. This command locates a record based on either its Message ID or its Message Delta Time fields.

2.4.3 Message Definition Index

a. The Message Definition Index operates on the following Message Description fields:

- System ID.
- Message ID.
- Message Description.
- Form Name.
- Message Verification.

b. The following details apply to the Message Definition Index commands:

(1) Append. This command fails if the concatenated key of System ID and Message ID already exists, or if the Form Name field does not reference the INGRES form for that message. In addition to adding a new Message Description record, this command generates a set of Message Composition records with default values, based on the content of the INGRES form. The Message Verification field of the new Message Description record is set to indicate an incomplete format definition.

(2) Delete. This command deletes the specified Message Description record and its corresponding Message Composition records. The delete fails if this message format has been used to define any event.

(3) Copy. This command copies the specified Message Description record and its corresponding Message Composition records. The new Message ID and Message Description fields are operator-entered.

(4) Specify. This command provides access to the data base functions operating on the Message Composition records, through the Message Definition Specification menu.

(5) Find. This command locates a Message Description record, based on it's System ID and Message ID fields.

(6) Verify. This command checks all Message Composition records of the specified message, making sure all fields are valid. The Message Verification fields of the Message Description record is modified to indicate the status of the message format definition.

2.4.4 Message Definition Specification

a. Messages are described in terms of their field structure through the Message Definition Specification menu. The following Message Composition fields are visible to the operator:

- Field ID.

- Form Field.

- Field Type.

- Field Address.

b. The following details refer to the Message Definition Specification menu commands:

(1) Append. The operator is warned if either the Form Field or Field ID references are undefined. This command fails if this message has been used to define one or more events.

(2) Replace. The same restrictions on the append command apply to this command.

(3) Delete. This command fails if this message has been used to define any event.

(4) Find. This command locates a field based on its Field ID or Field Address fields.

2.4.5 Field Definition Index

a. Fields are defined through the Field Definition Index menu, which operates on the following Field Conversion fields:

- System ID.

- Field ID.

- DB Type.

- Conversion Type.

- Conversion Length.

- Field Description.

b. The following details refer to the Field Definition Index menu commands:

(1) Append. This command fails if the System ID and Field ID do not form a unique key. If the Conversion Type field indicates a conversion table, the specify command is automatically invoked. This command fails if this field is referenced in any message definition.

(2) Replace. This command fails if this field is referenced in any message definition or if the System ID or Field ID fields are modified.

(3) Delete. This command deletes the specified Field Conversion record and all associated Conversion Table records. The delete fails if the field has been referenced in any message definition.

(4) Copy. This command generates a new Field Conversion record, using operator-entered Field ID and Field Description fields. In addition to creating a new Field Conversion record, any associated Conversion Table records are generated.

(5) Specify. This command accesses the Field Definition Specification menu which is used to specify ASCII-to-bit-value conversions. This command fails if the Conversion Type field does not indicate such a conversion.

(6) Find. This command locates a Field Conversion record based on its System ID and Field ID.

2.4.6 Field Definition Specification

a. The Field Definition Specification menu operates on the following Conversion Table fields:

- ASCII.

- Bit Value.

b. The following details refer to the Field Definition Specification menu commands:

(1) Append. This command fails if a duplicate ASCII value is found. The operator is warned if this field has been used to define any message.

(2) Replace. The same restrictions on the append command apply here.

(3) Delete. The command fails if an attempt is made to delete the last entry in the table.

APPENDIX E

DISTRIBUTION

DISTRIBUTION LIST

Addressee	Number of Copies
Commander	
US Army Test and Evaluation Command	
ATTN: AMSTE-AD-M	3
AMSTE-TO	2
AMSTE-AD-A	1
AMSTE-CT	3
AMSTE-CM	3
Aberdeen Proving Ground, MD 21005-5055	
Commander	
Defense Technical Information Center	
ATTN: DTIC-DDR	2
Cameron Station	
Alexandria, VA 22314-5000	
Commander	
US Army Aberdeen Proving Ground	
ATTN: STEAP-MT-M	2
Aberdeen Proving Ground, MD 21005-5000	
Commander	
US Army Yuma Proving Ground	
ATTN: STEYP-MSA	2
Yuma, AZ 85364-5000	
Commander	
US Army Jefferson Proving Ground	
ATTN: STEJP-TD-E	1
Madison, IN 47250-5000	
Commander	
US Army Dugway Proving Ground	
ATTN: STEDP-PO-P	1
Dugway, UT 84022-5000	
Commander	
US Army Cold Regions Test Center	
ATTN: STECR-TM	1
APO Seattle 98733-5000	
Commander	
US Army Electronic Proving Ground	
ATTN: STEEP-TM-AC	4
Fort Huachuca, AZ 85613-7110	

PREVIOUS PAGE IS BLANK

Addressee	Number of Copies
Commander	
US Army Tropic Test Center	
ATTN: STETC-TD-AB	1
APO Miami, FL 34004-5000	
Commander	
US Army White Sands Missile Range	
ATTN: STEWS-TE-PY	4
STEWS-TE-O	1
STEWS-TE-M	1
STEWS-TE-A	1
White Sands Missile Range, NM 88002-5000	

DATE

FILMED

0 - 8